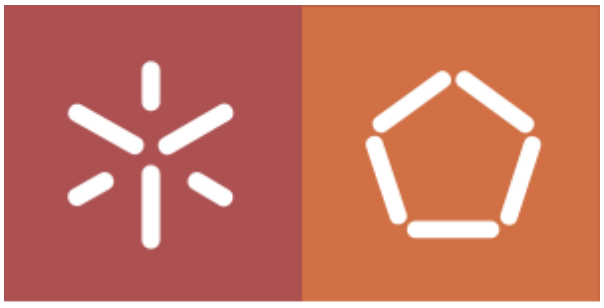
Sistema de Troca Segura de Mensagens

Universidade do Minho	Licenciatura em Telecomunicações e Informática Criptografia e Segurança em Redes	José Novais - N.º de aluno A105056 Miguel Machado - N.º de aluno A103668 Tiago Diogo - N.º de aluno A103665



Universidade do Minho
Escola de Engenharia

Introdução

Este trabalho tem como objetivo o desenvolvimento de um sistema assíncrono para troca segura de mensagens entre clientes, composto por um servidor central e vários clientes. O sistema é projetado para garantir a confidencialidade, integridade e autenticidade das mensagens, permitindo que dois ou mais clientes se comuniquem de forma segura através de um servidor intermediário. A importância da segurança neste contexto deve-se ao risco de ataques, como interceção e modificação de mensagens, sendo essencial garantir que as comunicações não sejam adulteradas ou vistas por terceiros.

Descrição Geral do Sistema

- O sistema consiste em um servidor e vários clientes que trocam mensagens utilizando o servidor como intermediário. O servidor armazena mensagens que só serão entregues ao destinatário quando este solicitar. Cada mensagem inclui os campos `id_msg`, `id_origin`, `id_destination`, `subject`, `content` e `timestamp`. As mensagens são criptografadas e assinadas digitalmente, garantindo a segurança durante a troca das mesmas.

- O fluxo do sistema pode ser visualizado da seguinte forma:

Registo do Cliente: Cada cliente se regista fornecendo uma senha e gerando um par único de chaves RSA. A chave pública é armazenada no servidor.

Login: O cliente faz login utilizando a sua senha, e um token de sessão é gerado. Esse token é utilizado nas comunicações subsequentes para autenticação.

Troca de Mensagens: As mensagens são criptografadas com AES e assinadas digitalmente pelo remetente, garantindo que apenas o destinatário possa lê-las e que a origem da mensagem possa ser verificada.

Requisitos de Segurança

Os requisitos de segurança incluem:

- Confidencialidade: As mensagens devem ser criptografadas para garantir que apenas o destinatário possa lê-las.

- Integridade: Deve-se garantir que as mensagens não sejam alteradas durante o armazenamento no servidor.

- Autenticidade: Os destinatários devem ser capazes de verificar a identidade do remetente.

Ferramentas Criptográficas Utilizadas

Para atender aos requisitos de segurança, utilizamos uma combinação de criptografia simétrica (AES) e assimétrica (RSA), bem como assinaturas digitais.

- AES no Modo GCM (Confidencialidade e Integridade):

Utilizamos o AES (Advanced Encryption Standard) no modo GCM (Galois/Counter Mode) para criptografar o conteúdo das mensagens. AES é um algoritmo de criptografia simétrica muito eficiente e seguro, amplamente utilizado para proteger dados sensíveis.

O modo GCM do AES fornece confidencialidade (garantindo que apenas o destinatário possa ler a mensagem) e integridade (garantindo que o conteúdo não seja alterado sem que isso seja detetado). Além de criptografar os dados, o GCM gera uma tag de autenticação que permite validar se o conteúdo foi modificado. Isso garante que qualquer tentativa de alteração seja detetada.

- RSA para Troca Segura de Chaves (Segurança da Chave Simétrica):

Utilizamos a criptografia assimétrica RSA (Rivest–Shamir–Adleman) para proteger a chave simétrica usada no AES. Como o AES utiliza uma chave única para criptografar e decifrar, é essencial proteger essa chave durante a sua transmissão.

RSA é particularmente útil para a troca segura de chaves. Cada cliente possui um par de chaves RSA (pública e privada). A chave pública do destinatário é utilizada para criptografar a chave AES antes de ser enviada, garantindo que apenas o destinatário, com a sua chave privada correspondente, possa decifrar essa chave e, portanto, o conteúdo da mensagem. Isso evita que um intermediário tenha acesso à chave AES e ao conteúdo protegido.

- Assinaturas Digitais (Autenticidade e Integridade):

Utilizamos assinaturas digitais para garantir a autenticidade das mensagens. Cada cliente assina digitalmente o conteúdo da mensagem com sua chave privada antes de enviar. O destinatário pode então verificar a assinatura utilizando a chave pública do remetente.

Essa abordagem garante não apenas que a mensagem veio de quem diz que enviou (autenticidade), mas também que o conteúdo não foi alterado desde a sua criação (integridade). Isso ajuda a proteger contra ataques em que alguém tenta modificar a mensagem durante a transmissão.

- Bcrypt para Hashing Seguro de Senhas (Autenticação):

Durante o registo, a senha do cliente é "hasheada" utilizando bcrypt, que é um algoritmo de hashing desenhado para ser computacionalmente dispendioso, dificultando ataques de força bruta.

O uso de bcrypt adiciona um salt (valor aleatório que é combinado com a senha original antes de ser gerada a sua versão criptografada) único para cada senha, tornando ataques por dicionário mais difíceis e ajudando a proteger senhas mesmo que a base de dados seja comprometida.

- Certificados SSL/TLS

Utilizamos certificados SSL/TLS autoassinados para proteger a comunicação entre clientes e servidor. Estes certificados proporcionam uma camada básica de proteção contra ataques do tipo man-in-the-middle.

Autenticação e Autorização

- Registo e Login: Cada cliente deve se registar fornecendo uma senha, que é armazenada no servidor usando o bcrypt para hashing. No login, um token de sessão é gerado e usado para autenticação nas comunicações subsequentes.

- Validação de Tokens: O servidor verifica a validade dos tokens de sessão para garantir que apenas utilizadores autenticados possam enviar ou consultar mensagens.

Testes de Segurança e Resultados

- Realizamos vários testes para validar os aspectos de segurança do sistema:

Teste de Integridade: Alteramos o conteúdo de uma mensagem diretamente na base de dados do servidor. Quando o destinatário tentou ler a mensagem, o cliente falhou na validação da assinatura digital, indicando que o conteúdo havia sido adulterado. Esse teste confirmou que o sistema é capaz de detectar modificações indevidas, garantindo a integridade das mensagens.

Teste de Confidencialidade: Foi realizado um teste para garantir que apenas o destinatário pudesse decifrar as mensagens. A chave AES, criptografada com a chave pública do destinatário, garantiu que o conteúdo não pudesse ser acessado por terceiros.

Teste de Autenticidade: Testamos a autenticação utilizando assinaturas digitais. Apenas o destinatário com acesso à chave pública do remetente foi capaz de verificar a assinatura e confirmar a origem da mensagem.

Tratamento de Exceções e Robustez

Para melhorar a robustez do sistema, implementamos tratamento de exceções tanto no cliente quanto no servidor. Por exemplo, ao tentar decifrar uma mensagem ou interpretar uma string em base64, o cliente informa ao utilizadore caso haja falha, indicando que a mensagem pode ter sido adulterada. Isso é fundamental para garantir uma boa experiência ao utilizador e a segurança do sistema.

Algumas funções importantes:

- Geração de Certificados (linhas 26-71 - Servidor):

generate_self_signed_cert() - Responsável por gerar o certificado e chave do servidor. Isso é essencial para garantir que as comunicações entre o servidor e os clientes sejam feitas de forma segura (TLS).

- Registo e Login de Clientes (linhas 133-199 - Servidor):

handle_register() - Cuida do registo de novos clientes.

handle_login() - Responsável pelo login dos clientes já registados. A senha é comparada utilizando bcrypt para garantir segurança na autenticação.

- Envio de Mensagens (linhas 200-273 - Servidor):

handle_send() - Garante o envio seguro das mensagens, validando assinaturas digitais e armazenando as mensagens criptografadas. Nesta função, os elementos de segurança como assinatura e criptografia são aplicados para garantir a integridade e confidencialidade.

- Geração ou Carregamento de Chaves RSA (load_or_generate_keys(), linhas 16-47 - Cliente)

Esta função garante que cada cliente tenha um par de chaves RSA (pública e privada). Caso as chaves já existam, elas são carregadas; caso contrário, novas chaves são geradas e guardadas em arquivos localmente.

Importância: A existência de um par de chaves RSA é fundamental para a criptografia assimétrica usada no sistema. A chave pública é compartilhada com o servidor e outros clientes, enquanto a chave privada é usada para operações de assinatura e decifrada.

- Criptografia e Decifragem de Conteúdo (AES) (encrypt_content(), decrypt_content(), linhas 49-76 - Cliente)

encrypt_content() usa AES no modo GCM para criptografar o conteúdo das mensagens.

AES (Advanced Encryption Standard) é um algoritmo simétrico usado para proteger a confidencialidade dos dados.

decrypt_content() é responsável por decifrar as mensagens recebidas, utilizando a chave AES e os parâmetros (nonce e tag).

Justificação e Vantagem: O AES é rápido e seguro para a criptografia de grandes volumes de dados, e o modo GCM (Galois/Counter Mode) garante tanto a confidencialidade quanto a integridade do conteúdo, gerando uma tag de autenticação que é verificada durante a decifragem.

- Criptografia e Decifragem da Chave AES (RSA)

(encrypt_key_with_recipient_public_key(), decrypt_key_with_private_key(), linhas 77-100 - Cliente)

encrypt_key_with_recipient_public_key() criptografa a chave AES usando a chave pública do destinatário para garantir a troca segura da chave simétrica.

decrypt_key_with_private_key() permite ao destinatário recuperar a chave AES com sua chave privada.

Justificação e Vantagem: A criptografia assimétrica RSA é usada para proteger a chave AES. Este processo garante que a chave simétrica (que é usada para criptografar o conteúdo da mensagem) seja transmitida de maneira segura, impedindo que um terceiro consiga acessar o conteúdo, mesmo que intercepte a mensagem.

- Assinatura Digital e Envio de Mensagens (send_message(), linhas 210-271 - Cliente)

Antes de enviar a mensagem, o cliente assina o conteúdo criptografado com sua chave privada utilizando o algoritmo RSA e o modo de preenchimento PSS (Probabilistic Signature Scheme) . Em seguida, a mensagem, a assinatura e a chave criptografada são enviados ao servidor.

Justificação e Vantagem: A assinatura digital garante a autenticidade e integridade do conteúdo. O destinatário pode verificar a assinatura usando a chave pública do remetente. Isso confirma que a mensagem foi realmente enviada por quem diz ter enviado e que não foi alterada durante o envio.

- Leitura e Validação de Mensagens Recebidas (read_message(), linhas 314-383 - Cliente)

Ao ler uma mensagem, o cliente primeiro decifra a chave AES, usando a chave privada do destinatário, e depois decifra o conteúdo da mensagem.

Em seguida, a assinatura digital é verificada usando a chave pública do remetente para garantir a integridade e autenticidade do conteúdo.

Justificação e Vantagem: Ao realizar a verificação da assinatura digital, garantimos que a mensagem não foi adulterada e que foi realmente enviada pelo remetente correto.

Considerações sobre Segurança Adicional

- Utilizamos certificados SSL/TLS (autoassinados) para proteger a comunicação entre clientes e servidor. Embora não sejam tão seguros quanto certificados emitidos por uma Autoridade Certificadora (CA) confiável, ainda assim proporcionam uma camada básica de proteção contra ataques do tipo man-in-the-middle, onde um atacante tenta intercetar a comunicação.
- O uso do SSL/TLS garante que as comunicações entre clientes e servidor sejam criptografadas, impedindo que terceiros visualizem os dados trocados.
- Melhorias Futuras: No futuro podemos implementar o uso de certificados emitidos por uma autoridade confiável.

Conclusão

O sistema desenvolvido atende aos requisitos de segurança propostos, garantindo a confidencialidade, integridade e autenticidade das mensagens trocadas entre os clientes. As ferramentas criptográficas escolhidas foram eficazes na proteção das mensagens, e os testes

realizados confirmaram a capacidade do sistema de detectar alterações indevidas. Durante o desenvolvimento, aprendemos a integrar diferentes técnicas de criptografia e a lidar com os desafios de segurança. Como trabalhos futuros, propomos melhorias na gestão de certificados, na autenticação dos utilizadores e na detecção de anomalias.

Referências

Bibliotecas Utilizadas: `cryptography`, `bcrypt`, `ssl`.

Documentação Oficial: Documentação das bibliotecas usadas e guias de criptografia moderna.

Artigos sobre Segurança: Leituras adicionais sobre criptografia assimétrica, simétrica e SSL/TLS.

Exercícios desenvolvidos durante as aulas teórico-práticas.